

# Fuzzy logic simplifies complex control problems

Tom Williams  
Senior Editor

***Fuzzy logic is proving a powerful methodology for control applications. Over 2,000 patents have been issued in Japan. Now, development tools and hardware are becoming available in the United States, and application possibilities abound.***

**O**n close examination, "fuzzy logic" turns out to be like Fuzzy Wuzzy in the nursery rhyme, not so fuzzy after all. The inputs, calculations and outputs of fuzzy logic all involve precise numbers handled in a mathematically rigorous manner. What's vague about fuzzy logic is the linguistic expression of a problem and its solution, not the numeric representation. But a vague or general linguistic expression of a problem can be very powerful. It lets users develop and refine a numeric representation without requiring them to develop or even understand a detailed numeric model. In fact, fuzzy logic can now be applied to complex logical and combinatorial problems for which it's impossible to build numeric models because of the enormous number of possible combinations.

It's said that fuzzy logic deals with vagueness and ambiguity, and this is true. It deals with relations between fuzzy sets, first described by professor Lotfi Zadeh of the University of California at Berkeley. Consider whether a man of 50 is old, for example (see "Fuzzy logic works with degrees of truth," p 98). The fuzzy set OLD MEN might appear to be arbitrarily drawn, but a some-

what different curve could with equal justification represent OLD MEN. Different ages would yield slightly different degrees of membership. Such differences reflect the expert knowledge or research results that define the set. And not all experts agree 100 percent.

The mathematical precision of fuzzy sets derives from the precise mapping of input values to degrees of membership. The linguistic power of fuzzy logic lies in its ability to define and manipulate sets that contain varying degrees of membership without having to deal in detail with all the combinations they can produce. The descriptive power of fuzzy logic comes from its ability to let experts express their knowledge in language they understand.

## ■ How fuzzy logic works

The use of adjectives to describe a problem is one key to fuzzy logic's ability to accommodate ambiguity: adjectives describe an application's fuzzy aspects. For example, what does a person mean when using terms like "small," "large" or "fast"?

Membership functions and rules provide the ability to handle complex combinations easily, which is

an important benefit of fuzzy logic. An application's rules and membership functions also contain the expert's knowledge about a system.

The adjectives used to formulate rules are more rigorously defined in an application's membership functions. The shapes of the membership functions can be changed, though "experience has shown that a relatively small number of triangular and trapezoidal membership functions is adequate for many control applications," according to David Brubaker, president of the Huntington Group (Menlo Park, CA) consulting firm.

When a processor scans an application's rule base, it tests each rule to determine whether its IF conditions have been satisfied. When the IF conditions are met, execution branches to the rule's THEN path, and the rule is said to have "fired."

Although a control application's logic may be fuzzy, the measured inputs from a physical system, and the outputs required to control it, will be precise. Precise values in fuzzy logic systems are termed "crisp."

In most cases, several rules fire and contribute to the output. The output values must then be resolved to yield a crisp value. Current prac-




 bringing the best together...

Most vendors lock you into their framework's point solutions front to back, like an electronic straightjacket. It's a take it or leave it, all or nothing approach, with no capability to insert whatever tools you prefer. That's why our PowerFrame™ comes with an optional tool kit that opens up the design process to any tools you choose. And we offer it with our standard set of tools because they're the best in the business. Viewlogic's Workview Series for schematic capture and simulation. Our own SCICARDS for placement and routing. AutoCAD's drafting and documentation, and Thermax thermal analysis. A framework with a set of tools we integrate—the best from a variety of vendors—because nobody makes the best of everything. A framework offering the point solutions you want—ours, theirs, and yours—within a true framework integration available now on the Sun and Digital platforms. When we say "bringing the best together," the best is up to you.

SCICARDS is a registered trademark of Harris Corporation. Viewlogic and Workview Series are trademarks of Viewlogic Systems, Inc. AutoCAD is a trademark of Autodesk, Inc. Thermax is a trademark of Helios Software Engineering, Ltd. Sun is a trademark of Sun Microsystems, Inc. Digital and PowerFrame are trademarks of Digital Equipment Corporation.

©1991, Scientific Calculations, a division of Harris Corporation.



**HARRIS**  
SCIENTIFIC CALCULATIONS DIVISION

7796 Victor-Mendon Road  
P.O. Box H  
Fishers, NY 14453  
1-800-4-HARRIS Ext. 4438  
1-800-344-2444 (Canada)



One common means of “defuzzification,” obtaining a crisp output from a group of membership functions, is the centroid, or center of gravity, method. This requires overlaying the portions of output membership functions produced by several rules and calculating the center of gravity of the resulting shape as the final output value. Another popular method, called max height, involves selecting from a range of possible discrete output values the one that receives the greatest degree of belief from fuzzy processing. Yet another method is simply averaging the output values.

The simplicity of the pendulum problem's expression (see "An example of fuzzy logic control," p 93)—11 two-input rules and three sets of five membership functions—suggests that fuzzy logic can handle much more complex problems with far fewer rules than traditional, crisp expert systems. "Nuances in the interpretation of rules can be handled by degree of membership functions without having to deal





## ■ FUZZY LOGIC

with the detailed combinations that may arise," says Brubaker. "The result is that the number of rules goes way down." By relegating complex combinatorial problems to fuzzy sets, users are therefore allowed to express rules for the behavior of a system in the linguistic form that's closest to the way humans think.

The familiarity of the language used allows rapid description of problems, resulting in fast prototyping of fuzzy solutions. After establishing a "shell" (a minimal set of

tains a fuzzy logic database structure enabling it to recognize 3,535 hiragana, katakana and kanji characters. A built-in stylus lets users hand-write these characters on the screen. According to Hironobu Kawai of Sony's Supermicro Systems Group, the PTC-500 does it all with a 64-kbyte program in ROM and a ROM database of only 128 kbytes.

Complex kanji characters are classified into smaller groups of strokes. The strokes in those groups are then classified by their length,

gone through all its levels, there may still be various "leaves"—nodes with no further branch paths—that are assigned some degree of validity. Some may even represent the same character, since there are about 1 million leaves in the entire database. The leaf with the highest degree of belief is turned into a crisp value and selected as the character to be recognized.

## ■ Complex controls

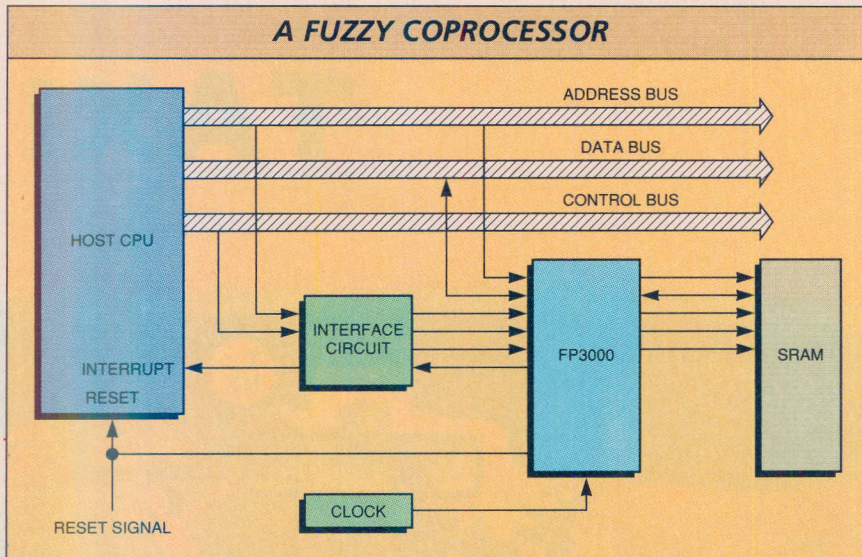
Some control applications may entail not only servo-type situations, such as the inverted pendulum, but also complex combinatorial problems, such as the efficient dispatch of elevator cars in a large building to minimize the individual waiting time for passengers. Mitsubishi Electric (Osaka, Japan) has applied fuzzy logic to elevator car dispatch and achieved a 15 to 20 percent reduction in average waiting time and a 30 to 40 percent reduction in long waits of 60 seconds or more.

The AI-2100 Elevator-Group control system makes use of two kinds of fuzzy rule sets, off-line rules and on-line rules. Off-line rules use fuzzy logic to decide which on-line rules should be applied. Off-line rules decide, among other things, where cars not in use should be parked to ensure efficient dispatch in highly variable traffic demands. On-line rules are applied when hall calls are registered—that is, when somebody pushes a button. They include rules that prevent "bunching" by assigning calls so that cars don't travel in groups. This may result in an individual call not getting the car that could actually be there the quickest, but the overall waiting time is greatly reduced. The rules used were created by experts in group supervisory control.

An example of a rule preventing bunching is "IF a call from the high zone (of a building) is registered AND the number of cars ascending is large, THEN select a car from among those already ascending using an evaluation rule."

This prevents assigning another idle car to the group of ascending cars and running the risk that a call from a lower floor made after the dispatch decision will go unanswered for an unacceptably long time.

Using fuzzy logic, the system designer can make additions to the rule base to improve decision mak-



*The FP3000 fuzzy processor from Omron has an interface similar to that of an SRAM for interfacing to various hosts. In expanded mode with an attached SRAM for rule base memory, the FP3000 can handle up to three sets of 128 rules each. The host can read and write the rule memory whenever the FP3000 is idling, but not while it's performing inferences.*

rules and membership functions that demonstrates feasibility), the designer can refine the system by experimenting with different rules and membership functions. Such prototyping is, of course, greatly aided if there's an adequate set of hardware and software development tools at hand.

## ■ Fuzzy recognition system

Another application embeds fuzzy processing inside a database search to perform character recognition. The problem of handwritten-character recognition has been a tough nut to crack for years, and that's just in recognizing Roman block printing, or at most 256 ASCII characters. Imagine the problem of recognizing handwritten Japanese characters.

The new PTC-500 palmtop computer from Sony (Tokyo, Japan) con-

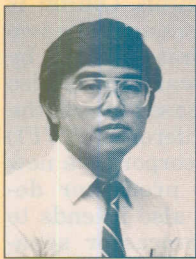
direction and relation to each other. The database is a tree structure, but the connections between nodes on the tree are fuzzy rules. A rule might state something like "IF line A is shorter than line B and parallel to B, THEN branch to the node that satisfies the greatest degree of belief in the conditions." That would be an example of max height defuzzification.

The attributes "shorter" and "parallel" are evaluated according to their membership in fuzzy sets. The database knows the standard forms of the characters and evaluates the stroke combinations for their degree of conformity to the standard.

Each stroke in a character takes the evaluation one level deeper in the tree-structured data. For every level down, a number of branches at that level may have a degree of validity. When the analysis has



## An example of fuzzy logic control



**A** classical example of how a fuzzy logic control system works is the problem of balancing an inverted pendulum, implemented here by Togai InfraLogic (Irvine, CA) in its

TILShell development environment. A weight at the end of a shaft is mounted on an electric motor that's used to balance it in an upright position, as shown in the center window. If the pendulum starts to fall to one side, current is applied to the motor in the opposite direction to bring the pendulum upright again. A closer look reveals that there are three factors of interest: The angle (theta) between the pendulum and vertical; the speed (delta theta) at which the pendulum is falling; and the amount of current to be applied to the motor.

Intuitive reasoning would try to express some rules governing the system. It would tell us that if the pendulum is just a little bit off-center, a small amount of corrective current should be applied. But if the pendulum is off by a large degree and its speed is increasing rapidly, a large amount of current should be applied to bring the pendulum back to a stable position.

The key point here is that adjectives have been introduced to the description of the problem. It's the adjectives that constitute the "fuzzy" elements. Just what is "a little bit" or "a large amount"?

The actual inputs to any physical system trying to solve this problem will be exact in terms of degrees and degrees-per-second. And only a precise current output will make the system behave properly. Such values in fuzzy systems are called "crisp."

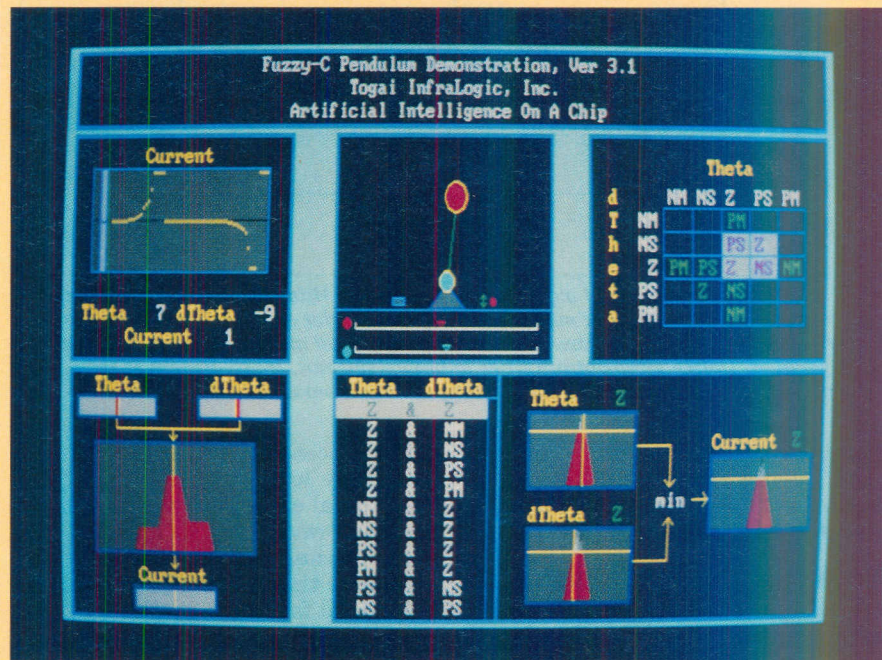
It's the rules that are general and the membership functions that are fuzzy because they cover complex combinations that otherwise would be unmanageable. The rules and membership functions are also the repositories of an expert's knowledge about a system. Once the expert writes "a little bit" in a rule, he or she must define what "a little bit" means in a membership function.

For the purposes of the pendulum problem, a set of five membership functions has been declared for each input: negative medium, negative small, zero, positive small, and positive medium. They can be shown graphically as a series of triangles and trapezoids that cover a range of -1 to 0 to +1. Actual input value ranges are normalized to this generalized range.

The next step is to become a little more linguistically rigorous and declare a set of formal IF-THEN rules, such as "IF angle theta is positive small and delta theta is zero, THEN current is negative small." Even though one member-

according to fuzzy logical operations, and various output membership functions are combined to produce a final crisp output. The lower center window shows the rule list with the membership functions of the highlighted rule displayed in the lower right window.

In this example, the output is a value of current to the motor. The lower left window dynamically shows how varying input values affects the combined output membership functions, and how those are "defuzzified" with the center-of-gravity method. The upper left window graphs the magnitude and direction of current to the motor.



ship function is called zero, there are still some small positive and negative values that belong to it. It could just as easily have been called "center." The upper right window shows the rule matrix. The inputs (IF) are along the axes of the matrix, and the outputs (THEN) are at the intersections.

When the processor scans the rule base, some rules "fire" according to the input conditions. In the screen's upper right window, rules that fire are highlighted in gray. When a rule fires, its IF conditions have been met, and execution branches to the THEN path of the statement. The rules are evaluated ac-

An inverted pendulum can be balanced with familiar binary logic. After all, the whole thing is carried out by a digital microprocessor system. But the natural language character of the problem's expression makes it easily understandable to persons whose primary expertise isn't computer science.

Fuzzy logic solutions are robust. In the pendulum example, the weight, the length of the shaft, and the strength of the motor can all be changed without altering the rules and membership functions. What changes is the scaling, or normalization, of input and output values.

**Maski Togai**, PhD, chairman and CEO, Togai InfraLogic



## FUZZY LOGIC

ing, without necessarily having to rewrite the program. For instance, since a hall button doesn't tell the system how many people are waiting on a floor, the elevator controller can add the factor of hall congestion to aid in car dispatch. A video camera compares its image of the hall and the people waiting with a stored image of the empty hall. Based on the image comparison, a membership value of empty, medium or

on the platform.

Most of the previous examples involve control situations that are "smooth" in the sense that they don't involve random interrupts that require fast context switches and deadline scheduling. A fuzzy logic system is supposed to behave so as to maintain the illusion that all the rules in its rule base are evaluated simultaneously. Of course, they're evaluated sequentially like any

processors, the FP3000 and the FP5000, were developed by Apt Instruments (Sunnyvale, CA) and then licensed to Omron. Omron is expected to offer versions of the two processors in the United States by the middle of the year. Apt is also developing a VME-based board, the truth value flow inference (TVFI) module, that will incorporate a new proprietary fuzzy processor designed by Apt. Apt also intends to market the new processor separately in the United States.

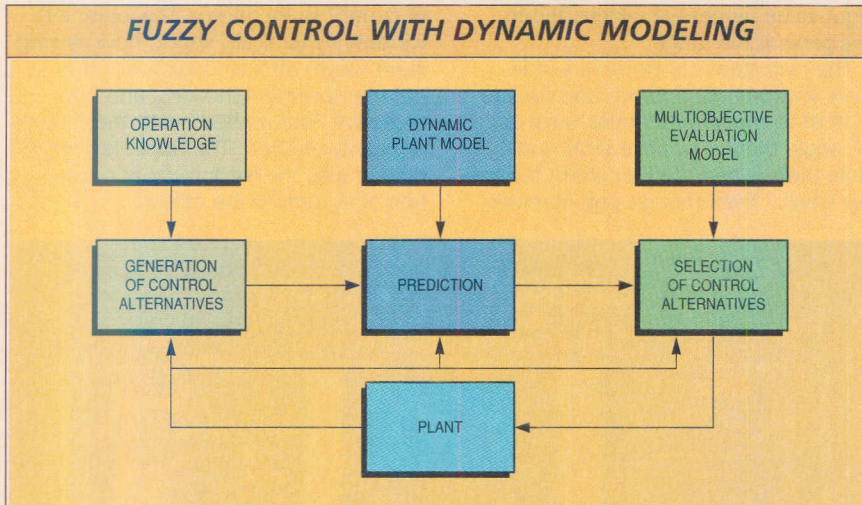
The Togai FC110 is a 10-Mips CMOS RISC processor with a specialized instruction set for fuzzy rule base evaluation. These include six instructions for fuzzy rule evaluation, such as LHS for "left-hand side," which evaluates the IF portion of a rule. The RHSC instruction evaluates the THEN portion of a rule, using the center-of-gravity method for defuzzification. FZAND and FZOR instructions perform fuzzy AND and OR (min./max.) operations.

Rules, membership functions and the FC110's program instructions reside in a knowledge base memory (KBM). The KBM can be EPROM, RAM or ROM and have a maximum size of 64,000 16-bit words. Inputs and outputs are 8-bit precision, and membership functions have 8 bits of resolution as well. Depending on the complexity of the rules, the KBM can handle up to 800 rules with their membership functions, and the FC110 can perform up to 100,000 rule evaluations per second.

The Omron FP3000 is a 24-MHz (max.) part that's intended to run in a system as a coprocessor to a host microprocessor. It can execute 100 single-antecedent, single-consequence rules, or 20 five-antecedent, two-consequence rules in 650 ms. Rules can have up to eight antecedents and two consequences and can be arranged in up to three groups of up to 128 rules each. The FP3000 has an interface for external SRAM, which is where the rules and membership functions are stored. Membership functions have a resolution of 12 bits, so each function can associate 4,096 values with input variables.

### Hybrid control strategies

This type of coprocessor arrangement for both the Togai and the Omron processors could lend itself to design of mixed fuzzy and crisp real-time systems. In such a system



*In the predictive fuzzy control algorithm developed by Hitachi, control alternatives are gauged against a dynamic model of the system under control. The results are then evaluated according to how well they fit an overall model that may have multiple objectives. Unlike interrupt priorities in binary systems, these objectives may have priorities that are also characterized by a certain amount of fuzziness.*

heavy congestion is established and used by the fuzzy rule base as a factor in the decision to dispatch elevator cars.

### Control with multiple goals

Another way of handling a system with multiple goals was developed by Hitachi's System Development Lab (Kawasaki, Japan) in an intelligent train control. Starting and stopping a train smoothly and efficiently is a task that must achieve many objectives—for instance, running time, energy conservation, safety, comfort, and stopping accuracy—and some of them may not be entirely compatible. Hitachi has developed a predictive fuzzy logic system that evaluates the effects of several control commands to select the one that most satisfies all the system requirements. The result is a subway train in Sendai, Japan that's legendary for its ride and efficiency. Passengers don't even need hand grips when the train starts and stops within 1.5 cm of a mark

other processor code.

A fuzzy logic system differs from traditional expert systems in that it has far fewer rules to evaluate. So the time constraint on a fuzzy-based system is that it can't be allowed to change so fast that control can't be maintained within the time the processor takes to scan the rule base. Thus, for more complex systems, there will be a constant push for more processor power and speed. Complex real-time systems characterized by large numbers of random interrupts probably don't lend themselves to fuzzy solutions yet.

### Fuzzy logic processors

Given the advantages of hardware assist, it's not surprising to find that several companies are offering dedicated fuzzy processors. The most prominent are Togai InfraLogic (Irvine, CA) and Omron (Kyoto, Japan). Togai is marketing its FC110 processor as a separate part and as an integrated set of up to four processors on its FCA10VME accelerator board. Two dedicated fuzzy



the host CPU could accept and handle interrupts caused by external events while monitoring and controlling other functions using fuzzy logic. It's even possible for an external event to cause an interrupt that the CPU would service by invoking one or another fuzzy rule set to be run on the coprocessor as long as the system's time constraints could be guaranteed. Of these processors, the Togai FC110 is currently shipping in the United States; the Omron parts are expected to be available by second quarter 1991.

It should be noted that comparing performance figures for fuzzy processors is such a new game that the figures are truly fuzzy. Much depends on the kinds of rules—whether they have one, two, or more inputs and one or two outputs. The resolution of I/O and of membership functions also influences the “rules-per-second” performance claims, as does the kind of membership function used. A membership function represented by a triangle or trapezoid (that is, straight lines) where only the height, width and slope of the lines may be specified lends itself to interpolation or implementation as a lookup table. A curved membership function defined by an equation will necessarily take longer to compute than a triangular function.

A fuzzy logic system need not be digital. Apollo Electronics (Fukuoka, Japan) has developed a machine that's now used to evaluate the results of orthodontic treatment, but this machine is capable of being adapted to many specialized applications. It allows inferences to be made of pre- and post-treatment conditions compared with a more or less ideal norm. The ideal norm isn't achievable in all patients, and is based on the subjective knowledge and experience of experts, but an assessment of how close one has come and how good a job has been done can be made.

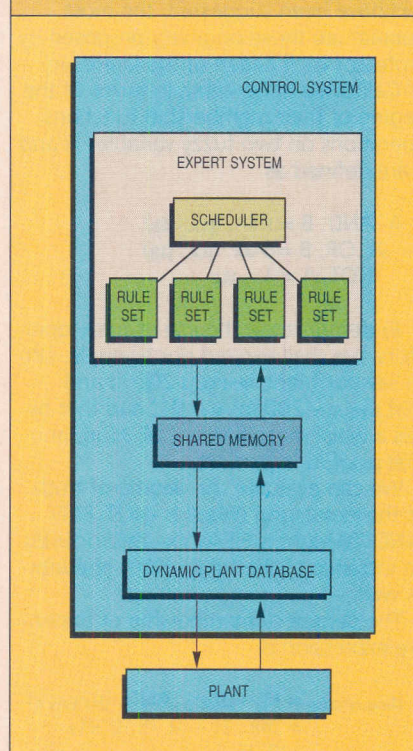
The Apollo machine uses a circuit board for each tooth, and membership functions have been set as analog values by adjusting variable resistors. This makes it easy to change the membership functions as well, since each board contains its own rule set and membership functions. The “before” and “after” values are entered with potentiometers, and an assessment is given for each tooth and then an overall assessment is inferred. This same technique used for such a specialized application as orthodontics has potential for many

types of complex analysis that require expert knowledge and for which there isn't an absolute norm.

### ■ Fuzzy development tools

Engineers with real application problems need more than new, unfamiliar processors. They require board-level products and software development environments to develop and prove design concepts and

#### HYBRID CONTROL SYSTEM



*In the expert system developed by Mitsubishi Electric, Boolean logic-based rules are used by the scheduler to focus on fuzzy rule sets to apply to given situations. The system interacts with the plant model via shared memory, so it can easily be applied in distributed configurations.*

build first-generation products. “Fuzzy logic is applied in the United States today, but only by engineering companies such as Apt and Togai who do projects for customers,” says Xiwen Ma, vice-president of R&D for Apt Instruments. “The availability of tools and hardware should make fuzzy logic available to general engineering.”

Development tools for fuzzy logic are available, and more are on the way, both from the United States and from Japan. Omron, for instance, has been offering its digital signal processor-based FZ3000 fuzzy logic board with the FT6100

support tool in Japan for some time. The FT6100 runs on the NEC 9801 personal computer, and so would require some modification for the U.S. market. Mitsubishi Atomic Power Industries (Tokyo, Japan) has a development toolset, called Fuzzic-1, that's written in QuickBasic for the NEC 9801. And the company has announced plans to convert Fuzzic-1 to C and port it to Unix for marketing in the United States.

Fuzzic-1 allows the user to graphically specify rules and membership functions and to simulate the behavior of the target software on-screen before compiling it to run on the target system. It's possible to watch shaded areas of membership function graphs change with input variables and to observe the effects on output membership functions. This allows the user to experiment with different rules and membership function specifications to try to optimize the application's performance.

### ■ Fuzzy CASE system

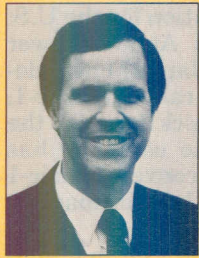
Togai currently offers its FC110-based VME board and a development environment, called TILShell, that runs under Microsoft Windows. TILShell could almost be called a fuzzy CASE environment. It starts out with a facility that lets the user graphically define the structure of a fuzzy system using input, output and processing objects that can then be assigned attributes. Attributes can define storage type, range of values, and membership functions.

TILShell also contains a membership function editor that lets the user graphically define the shape of a set of membership functions associated with a variable. Straight-line-type membership functions can be drawn with a mouse using the editor's point-and-click tools. If an application requires membership functions that are complex curves, such as a bell-shaped distribution curve, an equation editor lets the user specify the membership function using a set of math functions. A fuzzy editor works with a rule editor and a fragment editor to allow the user to define and edit rules and to add and delete rules and C code fragments to and from the rule base. The rule editor is menu-driven, while the fragment editor lets the user enter and edit C source code as text.

Once the fuzzy logic system has been defined, TILShell provides three options for generating code. The Fuzzy-C compiler generates C



## Fuzzy logic works with degrees of truth



**F**uzzy logic is an extension of traditional bilevel (or Boolean) logic. While Boolean logic requires a statement or a condition to be either completely TRUE or

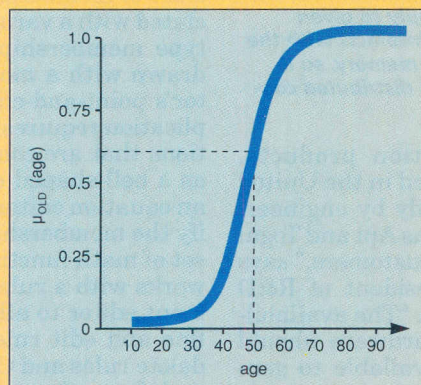
completely FALSE, fuzzy logic allows partial truth and partial falseness.

For example, consider a 50-year-old man. To say that he is OLD isn't entirely true, and yet we likewise can't say that he's NOT OLD. We're dealing with an in-between region, in which traditional logic has problems.

Turning to fuzzy logic, we can create a function relating age in years to oldness, and given this, we can say that the statement "the 50-year-old man is OLD" is 60% TRUE. The statement is also necessarily 40% FALSE. An assertion being simultaneously TRUE and FALSE is often a difficult concept for someone used to working with Boolean logic, but the world tends to be more gray than black-and-white.

Fuzzy logic is derived from the more-general theory of fuzzy sets. Using fuzzy sets to represent the previous example, we can say that the function relating age to oldness is a curve relating age to a degree of membership in the fuzzy set OLD. Degree of membership is typically designated  $\mu$ , and the degree of membership of the fuzzy set OLD as a function of age is given by  $\mu_{OLD}(\text{age})$ . For consistency with traditional logic,  $\mu$  ranges from 0 to 1, with 0 indicating null membership in the set and 1 indicating full membership.

A possible function for  $\mu_{OLD}(\text{age})$  is shown in Figure 1 (at right). The age 50 years is a member of the fuzzy set OLD with degree of membership  $\mu_{OLD} = 0.6$ . We might similarly say that the 50-year-old man is a member of the set OLD with degree of membership  $\mu_{OLD} = 0.6$ .



Fuzzy logic variables can be combined using operators similar to those for their Boolean cousins, although the operations are necessarily defined differently. The standard fuzzy operations are AND, OR and NOT, which correspond respectively to INTERSECTION, UNION and COMPLEMENT for fuzzy sets. Although many definitions of fuzzy logic operators have been suggested, the most popular are those originally proposed by professor Lotfi Zadeh of the University of California at Berkeley. In terms of the degree of their membership functions, operations on two fuzzy variables A and B are defined as

$$\begin{aligned} A \text{ .AND. } B &= \min. (\mu_A, \mu_B) \\ A \text{ .OR. } B &= \max. (\mu_A, \mu_B) \\ \text{.NOT. } A &= 1 - \mu_A \end{aligned}$$

To demonstrate the use of fuzzy operators, we shall expand our example. Let us say our friend weighs 200 lb and that  $\mu_{HEAVY}(200 \text{ lb}) = 0.91$ ; and that he has a waist measurement of 36 in. and that  $\mu_{FAT}(36 \text{ in.}) = 0.75$ .

We can evaluate the degree of truth of the expression (Waist is .NOT. FAT) .AND. (Weight is HEAVY)—this includes an instance of the fuzzy .NOT. operator as well.

The degree of membership of the entire expression is

$$\begin{aligned} \mu_{\text{expression}} &= (1 - \mu_{FAT}) \text{ .AND. } (\mu_{HEAVY}) \\ &= \min. (1 - 0.75, 0.91) \\ &= 0.25 \end{aligned}$$

This expression also sets the stage to introduce the dominant fuzzy control mechanism—the IF (condition) THEN (action) rule. Borrowed from expert systems, it requires that actions be executed if the condition part of the rule is at least partially true. But in a fuzzy system, this structure also has a powerful twist: the action is

executed with the degree of membership of the rule's condition. If a condition is only minimally true—say, for ex-

ample,  $\mu_{\text{condition}} = 0.1$ , then the action is executed with that same  $\mu$ ,  $\mu_{\text{action}} = \mu_{\text{condition}} = 0.1$ .

Returning to our example, we would like to devise some rules to help our friend lose weight. While both diet and exercise are appropriate, for this example we shall address only diet.

We will work with two inputs, Waist and Weight, and each will be represented by two fuzzy sets—NORMAL and FAT for Waist, and NORMAL and HEAVY for Weight. We shall define a single output, Diet, which will also have two fuzzy sets—WEIGHT LOSS and MAINTENANCE.

Given these, we can create a two-rule system to govern our friend's eating habits:

- IF (Waist is FAT .AND. Weight is HEAVY) THEN (WEIGHT LOSS)
- IF (Waist is NORMAL .AND. Weight is NORMAL) THEN (MAINTENANCE)

Figure 2 (facing page) shows how these rules would be applied to his current condition. The control flow of the figure starts in the lower left corner (the inputs) and is vertical as the inputs are translated into degrees of membership ( $\mu$ 's) in the input fuzzy sets. The flow then proceeds horizontally toward the right, as the inputs and their  $\mu$ 's are applied as conditions to the rules. Finally, the flow moves downward toward the lower right corner (the output), as the rule actions are translated into an actual crisp output.

The sequence may seem complex at first, but it's actually straightforward. Let's step through the single iteration shown in the figure:

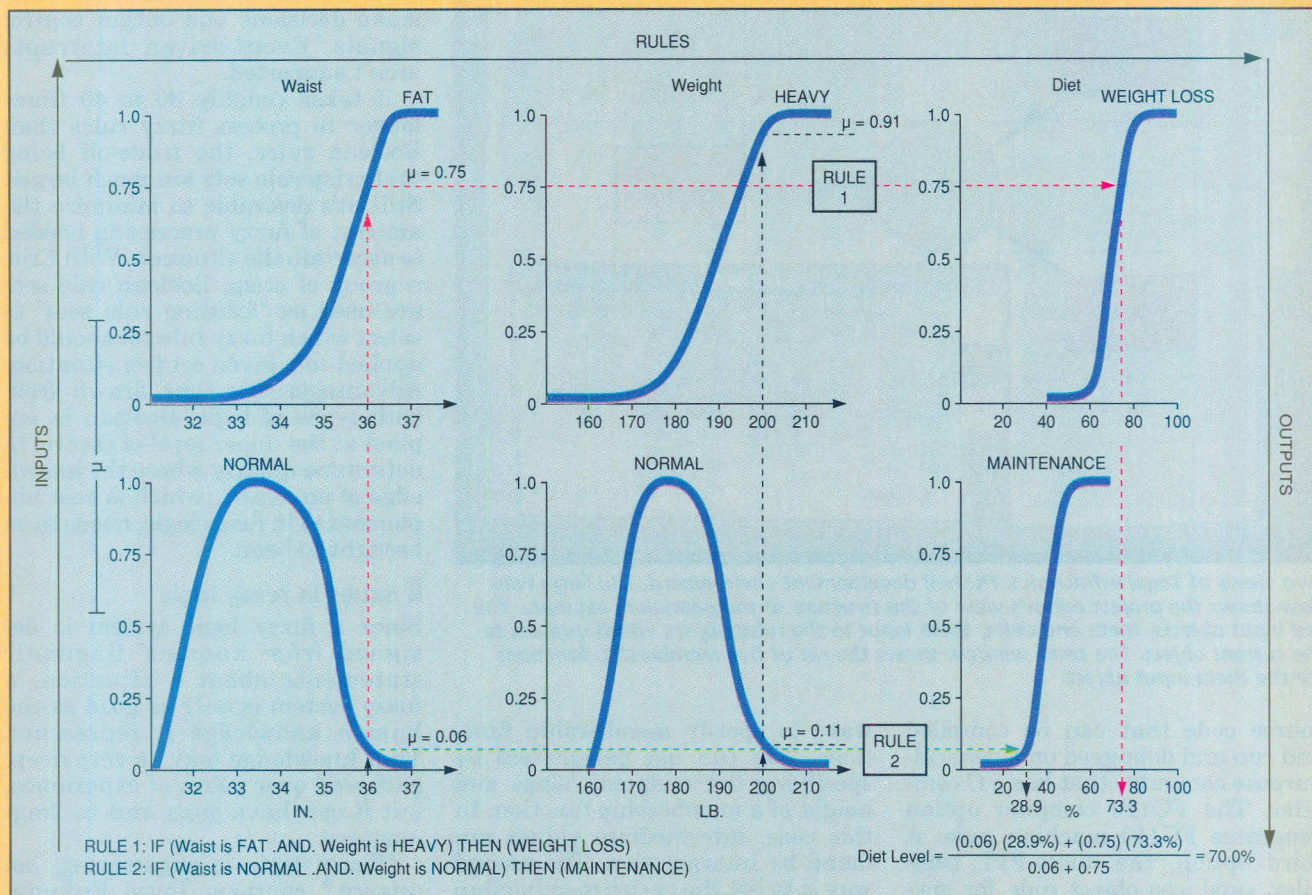
The current waist measurement, 36 in., is applied to the two fuzzy sets, FAT and NORMAL, for the input Waist, resulting in  $\mu_{FAT} = 0.75$  and  $\mu_{NORMAL} = 0.06$ . Our friend's waist is obviously more FAT than it is NORMAL.

His current weight, 200 lb, is applied to the two fuzzy sets, HEAVY and NORMAL, for the input Weight, resulting in  $\mu_{HEAVY} = 0.91$  and  $\mu_{NORMAL} = 0.11$ . Again, his weight is far more HEAVY than it is NORMAL.

The condition of Rule 1 requires a degree of membership ( $\mu$ ) of the expression (Waist is FAT .AND. Weight is HEAVY). The .AND. is performed using

David L. Brubaker, PhD, president, the Huntington Group





the minimum function, resulting in

$$\begin{aligned}\mu_{\text{expression}} &= \min. (\mu_{\text{FAT}}, \mu_{\text{HEAVY}}) \\ &= \min. (0.75, 0.91) \\ &= 0.75\end{aligned}$$

The  $\mu$  of the rule's condition is applied to the rule's action, in this case (for Rule 1) WEIGHT LOSS. Rule 1 results in a Diet level of 73.3% (in this example a 100% diet corresponds to fasting and 0% to complete gorging).

Similarly, for Rule 2 the condition requires the  $\mu$  of (Waist is NORMAL .AND. Weight is HEAVY).

$$\begin{aligned}\mu_{\text{expression}} &= \min. (\mu_{\text{NORMAL}}, \mu_{\text{NORMAL}}) \\ &= \min. (0.06, 0.11) \\ &= 0.06\end{aligned}$$

The  $\mu$  of Rule 2's condition is also applied to its action, MAINTENANCE. Rule 2 therefore results in a Diet level of 28.9%.

Finally, several techniques exist for combining these two outputs into a single, executable action. One of the more intuitive methods is to use an average, weighted by the respective degree of membership values. If we take this approach, the solution is

$$\begin{aligned}\text{Diet level} &= \frac{(0.06)(28.9\%) + (0.75)(73.3\%)}{0.06 + 0.75} \\ &= 70.0\%\end{aligned}$$

The required action is that our friend participate in a fairly heavy duty diet.

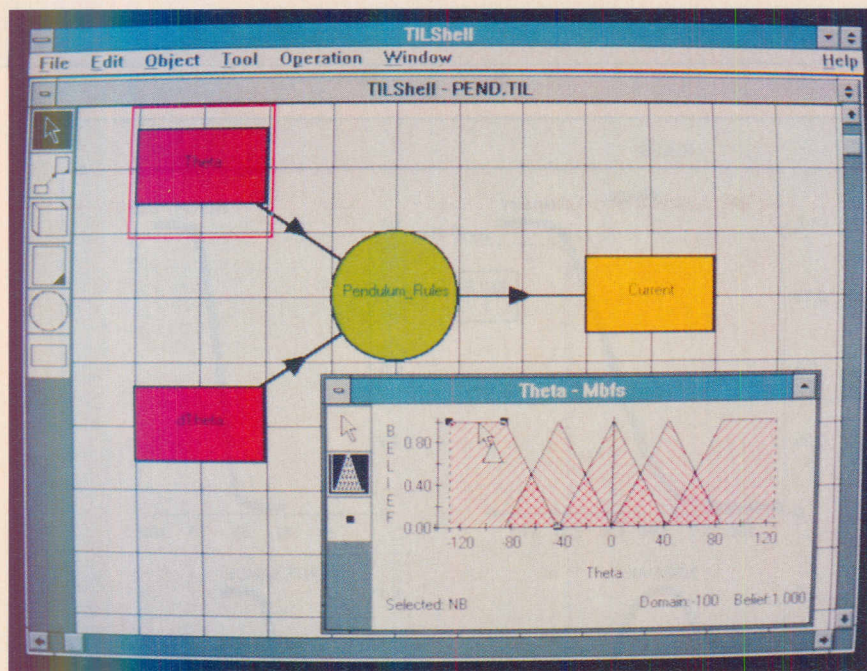
This has been a single iteration. To achieve the desired end result (shown by the two NORMAL curves to be a waist measurement of 33 to 34 in. and a weight of 180 lb), the rules will be executed on an on-going basis. As Diet (dominated by its WEIGHT LOSS component) is executed, both inputs (Waist and Weight) will start to decrease, resulting in lower  $\mu$ 's for both

FAT and HEAVY and higher  $\mu$ 's for both NORMALS. Ultimately Rule 1 will cease to have a significant effect, and Rule 2 will dominate, dictating execution of a MAINTENANCE diet. If the MAINTENANCE diet is set too low (allows too great a fat and caloric intake), our friend may start regaining his lost weight, and the WEIGHT LOSS component of Diet will again become increasingly active.

Surprisingly complex and powerful systems can be implemented by extending these basic techniques. The use of the fuzzy logic is especially appropriate, and often necessary, for complex systems, where an adequate system model is difficult or impossible to define; expert-controlled systems; systems with moderately to very complex continuous inputs and outputs; and systems where vagueness is common, such as those used in economics, natural sciences, and behavior sciences.



## ■ FUZZY LOGIC



Two views of Togai InfraLogic's TILShell development environment. The large window shows the project editor's view of the structure of the pendulum example. The red input objects, theta and delta, theta input to the rules object, which outputs to the current object. The small window shows the set of five membership functions for the theta input object.

source code that can be compiled and run and debugged on a general-purpose computer that has a C compiler. The FC110 compiler option generates FC110 machine code. A third option, the micro-FPL compiler, produces object code for several popular microprocessors, such as the MC 680X0 family and the H8.

### ■ Fuzzy design environments

For its new TVFI board, Apt Instruments is developing a software development environment called Fide (Fuzzy Inference Development Environment). Fide is expected to be available about the same time as the board (second quarter of 1991). The TVFI board is intended to operate in a system environment with a host CPU. At system boot-up, the application running on the host selects the appropriate fuzzy rule sets from disk files and loads them into the board's local memory. The host then places input data into input buffer space and reads outputs from output buffers in the board's memory.

Fide consists of a rule language, a compiler, a simulator, a debugger, and a graphical interface. The rule language is simple, according to Ma. To specify rules, you use simple IF-THEN constructs. There are two

ways to specify membership functions. You can list parameters by specifying the midpoint, slope and height of a membership function. In this case, intermediate values are found by interpolation. The second way is to list the entire membership function as a lookup table. Input value ranges are normalized to a zero-to-one scale, and membership

***A fuzzy system differs from traditional expert systems in that it has far fewer rules to evaluate.***



functions have 8 bits of resolution. The TVFI compiler compiles the rules and membership functions into code that the board's new processor can use.

Several Japanese companies have created in-house development environments for fuzzy application development. Besides Mitsubishi Atomic Power Industries, which created Fuzzic-1 for internal needs, Mitsubishi Electric has developed an environment, called Eric (Expert Real-time Intelligent Control), that

uses both crisp and fuzzy logic in a hierarchical structure. Systems designed with Eric use a fixed time cycle in which they take in data, make decisions and output control signals. Event-driven interrupts aren't supported.

It takes roughly 30 to 40 times longer to process fuzzy rules than Boolean rules, the trade-off being that crisp rule sets are much larger. Still, it's desirable to minimize the amount of fuzzy processing needed to ascertain the situation. With Eric, a group of crisp, Boolean rule sets are used as "focusing rule sets" to select which fuzzy rule set should be applied to a given control situation. Advantages are thus drawn from both types of logic: Boolean is applied at the upper level of control to determine quickly where the knowledge of an expert, which is best implemented in fuzzy logic, needs to be brought to bear.

### ■ Issues in fuzzy logic

Since a fuzzy logic system is designed from humans' linguistic statements about a situation, a fuzzy system is only as good as the human knowledge it represents. That knowledge may be very deep, gathered over years of experience, but it can have gaps and be fragmentary.

"Fuzzy logic is engineering, not science," cautions Toyoo Fukuda, manager of the advanced systems group for Mitsubishi Electric. "In control, for example, it's necessary to verify if the system is stable. We can't show that through fuzzy logic." In other words, fuzzy logic is still a young discipline and lacks the theoretical basis for proving its validity. The alternative to theory is empirical verification, which by its very nature can seldom be complete.

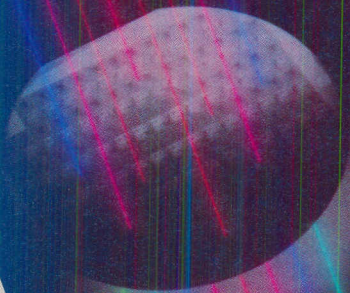
### ■ Reliability proven empirically

The lure of fuzzy logic is that by using human experience, it's fairly easy to get a system to seem to work. But a great deal of analysis, research and simulation is often required to get it to work well and to establish confidence in its reliability. For example, Matsushita (Osaka, Japan) has introduced various consumer products that incorporate fuzzy logic. These include a vacuum cleaner, rice cooker, a camcorder with image stabilization, a kerosene heater, and a washing machine. The washing machine has a single but-



INTERNATIONAL  
DESIGN ENGINEERING  
&  
ASIC EXPOSITION

idea



NINETYONE

complimentary  
EXHIBIT  
PASS

April 2 · 3 · 4, 1991

SAN JOSE  
CONVENTION CENTER

Free

**Exhibits,**

**Interactive  
Theater,**

**Product  
Showcases**

**and Panels.**

**15% Pre-Registration Discount**

**50 Technical and  
Management Training  
Sessions and Seminars.**

**ASiC**

Sponsored by ASiC Technology & News  
480 San Antonio Road, Suite 245  
Mountain View, CA 94040  
1.415.949.2742

**The year's most important event ...**

**for  
ASIC & EDA users.**

**Explore and analyze ...**

**every major  
ASIC & EDA company.**

**Learn and interact ...**

**with the industry's  
most experienced users.**

**Today's ...**

**issues and solutions.**

**Keynote ...**

**Scott McNealy**  
CEO, Sun Microsystems

**Register Now !**

**1-800-848-IDEA**

1 · 8 0 0 · 8 4 8 · 4 3 3 2

CIRCLE NO. 60



## ■ FUZZY LOGIC

ton. It's able to sense both the size of the load of clothes and the amount and type of dirt—whether it's soil- or oil-based—and automatically adjust the amount of soap and water and the washing time.

The load is sensed simply by filling a preset amount of water and turning the agitator a few times to measure the current drawn before starting the cycle. The amount and

Fuzzy Engineering (Yokohama, Japan), which was established by Japan's Ministry for International Trade and Industry and is supported by some 50 companies.

"The trade-off is easy design of a system at the sacrifice of optimum design; optimum means 100 percent," says Takeshi Yamakawa, director of the Fuzzy Logic Systems Institute, a privately funded re-

rules and membership functions for applications. A neural network has the combinatorial power to exercise all of the possible input combinations of a rule, and to show what the results will be. "The rule is implemented and run, and the neural network shows the results of all the different input conditions," says Fujiwara. "So you can shape the membership functions to correspond with what the results will be."

Matsushita's next likely step is to move the neural network from the development environment into the product. Through use, a product would "learn" to be more efficient and become more personalized by tuning its membership functions and, possibly, by eliminating unneeded rules from its rule base, resulting in faster performance.

Fuzzy logic applications appear to be ready to come of age in the United States. Tools, hardware and plenty of application examples are available. Some people, though, still have trouble with the name. When fuzzy logic was first introduced in Japan, it was called "aimai" (Japanese for fuzzy), and met considerable resistance. But the word "fuzzy," since it has no negative connotations, has become a marketing buzzword in Japan—it's been used in TV commercials for vacuum cleaners, for example. It's ironic that a control methodology based on linguistic principles met resistance due to a language quirk. Maybe we'd have better luck in the United States if we called it aimai. ■



*Matsushita's Yoshihiro Fujiwara feels that fuzzy logic is particularly suited to consumer products because of its ability to deal with nonquantifiable concepts. "Consumers want a good human/machine interface, so the nonquantifiable image or concept is familiar to the user of appliances incorporating fuzzy logic."*

type of dirt is sensed by a single optical sensor. The sensor and its logic measure how much the water is clouded, how long it takes for the clouding to reach saturation and what the level of saturation is to determine the type and degree of dirt. According to Yoshihiro Fujiwara, director of the intelligent electronics lab at Matsushita's central research laboratory, "You need to gather a lot of data and do a lot of simulation before you can determine these things." Once a prototype is developed, rules and membership functions must be refined by further simulation and analysis.

### ■ Verifying performance

There are two major theoretical problems facing fuzzy logic today, according to Fujiwara. One is that there is as yet no rigorous mathematical way of verifying the correctness of a fuzzy system. The second is that it's not currently possible to optimize the efficiency of a fuzzy system to an assured level of 100 percent. Exploring solutions to these problems is the main goal of the Laboratory for International

search foundation. "But if we can achieve 95 percent efficiency, that's very good, and it's easy to design at the sacrifice of that last 5 percent." So it's possible to know that a fuzzy system is working better than a conventional one, and even to measure how much better, but we still can't know exactly how good a fuzzy system can potentially be.

According to Yamakawa, simulation is the ordinary way to verify a system using fuzzy logic, but it hasn't been sufficient to gain the confidence of mathematicians and control engineers. "But some in the fuzzy field have recently begun to research the stability and reliability of fuzzy systems," says Yamakawa, "so it will be easier for mathematicians and control engineers to accept fuzzy logic, and that's an important step."

### ■ Fuzzy trends

The question is whether fuzzy verification can be solved with traditional equations. Both Yamakawa and Fujiwara see hope in the use of neural networks, which are being used by Matsushita to develop

*For more information about the technologies, products or companies mentioned in this article, call or circle the appropriate number on the Reader Inquiry Card.*

<b>Apollo Electric</b> 92-482-1300	Circle 209
<b>Apt Instruments</b> (408) 727-4910	Circle 210
<b>Hitachi/Systems Development Laboratory</b> 44-966-9111	Circle 212
<b>Matsushita</b> 03-3578-1237	Circle 213
<b>Mitsubishi Atomic Power Industries</b> 03-3433-8251	Circle 214
<b>Mitsubishi Electric</b> 06-497-7045	Circle 215
<b>Omron</b> 75-951-5111	Circle 217
<b>Sony/Supermicro Systems Group</b> 03-3448-4171	Circle 218
<b>The Huntington Group</b> (415) 325-7554	Circle 219
<b>Togai InfracLogic</b> (714) 975-8522	Circle 220